

Bag Reference Vector for Multi-instance Learning^{*}

Hanqiang Song^{*†}, Zhuotun Zhu[‡] and Xinggang Wang^{*}

^{*}Huazhong University of Science and Technology,

[†]University of Birmingham, [‡]University of California, Los Angeles

hanqiangsong1024@gmail.com, zhuotun@gmail.com, xgwang@hust.edu.cn

Abstract

Multi-instance learning (MIL) has a wide range of applications due to its distinctive characteristics. Although many state-of-the-art algorithms have achieved decent performances, a plurality of existing methods solve the problem only in instance level rather than excavating relations among bags. In this paper, we propose an efficient algorithm to describe each bag by a corresponding feature vector via comparing it with other bags. In other words, the crucial information of a bag is extracted from the similarity between that bag and other reference bags. In addition, we apply extensions of Hausdorff distance to representing the similarity, to a certain extent, overcoming the key challenge of MIL problem, the ambiguity of instances' labels in positive bags. Experimental results on benchmarks and text categorization tasks show that the proposed method outperforms the previous state-of-the-art by a large margin.

1. Introduction

Multi-instance learning (MIL), originally proposed for drug activity prediction [6], has been applied more frequently to diverse visual recognition tasks such as image retrieval, image classification, object detection, and visual tracking. In MIL, a typical weakly-supervised learning, training data are given as a form of labeled bags, each of which is composed of a wide diversity of instances associated with input features. The aim of MIL, in a binary task, is to train a classifier to predict the labels of testing bags, which is based on the assumption that a positive bag contains at least one positive instance while a bag is labeled negative if it is only constituted of negative instances. Thus, the crux of MIL is to deal with the ambiguity of instances'

labels, especially in positive bags which have plenty of cases with different compositions.

In a way, this weakly labeled instance framework cater to many existing visional tasks such as object recognition task for the reason that intrinsic structure of MIL is able to deal with some problems perfectly hence facilitate solutions. Take image classification for instance, an image is defined as a bag and patches in the image can be regarded as instances. Then according to the purpose of MIL, specific objects or key features can be defined as positive. By means of this MIL representation, crucial information can be captured.

Up to now, different algorithms have been designed to solve MIL problems. The previous methodologies are mainly in three folds: (1) Selecting key/discriminative instances and classifying bags based on the selected instances using generative or discriminative models, e.g., EM-DD [23], miSVM [1] and the key instance detection method [12]. (2) Mapping a bag into a high-dimensional feature space to get a vector representation of bag then training bag classifier, e.g. miFV [19]. (3) Constructing bag representation based on the internal structure of a bag - the relation between instances within a same bag, e.g., miGraph [24].

Different from the previous strategies, we aim to build bag-level representation based on the relative distance between bags. Take one bag for instance, other bags are regarded as reference bags functioned as the basis of the feature space. The derived bag representation is called bag reference vector (BRV). Then MIL task is transformed into a problem of classifying BRV. Thus, our method to solve this task is named miBRV (multi-instance learning of bag reference vector) as a whole.

Our motivation of proposing BRV is that it can capture the essential character of MIL. For supervised learning, the intra-class similarity should be higher than the inter-class similarity. In the same manner, the similarities between bags are distinctive features. To measure bag similarity, we use set-to-set distance which

^{*}This work is under the consideration of Pattern Recognition Letter.

considers all pairwise relations between two bags. We consider all pairwise distances between the instances in the compared bags, which is a typical set-to-set distance. In this paper, we extent Hausdorff distance [11] as the set-to-set distance measure. Furthermore, considering the ambiguity of instances in bags, we adopt a range of operators in Hausdorff distance to represent these relations.

The pipeline of generating bag feature is illustrated in Fig. 1. For every reference bag and operator, the input bag has a distance value to the reference bag based on the operator measurement. Total length of the bag reference operator is the product of the number of reference bag and the number of operators.

In the rest of this paper, we briefly review some related works in Section 2; then formalize the proposed miBRV method for MIL in Section 3; in Section 4, we carry out experiments of miBRV on MIL benchmarks and show the state-of-the-art performance; finally, we draw conclusions in Section 5.

2. Related Work

Multi-instance learning (MIL) has received a lot of attentions since it helps to solve a range of real applications. Till now, lots of MIL methods have been proposed to either develop effective MIL solvers or apply MIL to solve application problems. Firstly, we briefly review a few popular MIL solvers. The EM-DD method [23] uses EM to infer instance space with many instances from different positive bags and few instances from negative bags. Instead of adopting simple instance space, the miRPCA method [17] utilizes robust PCA model to build a instance model robust to outliers. Besides of generative instance models, discriminative models are more popular as instance model. For example, both MILBoost [16] and miSVM use discriminative methods, Boosting and SVM respectively, as instance models, and iteratively select positive instances to train models. Furthermore, miGraph [24] represents bag as graph and explicitly model the relationships between instances within a bag; while [5] models the relationships between different bags using conditional random field. Recent work [2] studies the problem if there are infinite number of instances in a bag.

MIL is useful for many computer vision applications. Originally, MIL is widely applied to image classification [13, 21], since it is able to exploit salient region in image where is critical for classification. In [9], a variant of miSVM called latent SVM is effective to find the parts of object for accurate object detection. Online MIL algorithms [3, 22] are popular for visual tracking. Recently, MIL has been widely used for weakly-supervised

object detection [18, 4].

In addition, our miBRV method is a reference-based method, which is analog to the popular concept ‘‘Attribute’’ [14] in computer vision. A face feature computed based on reference face is proposed in [15]. However, [15] is simpler and there is no MIL structure in it.

3. Multi-instance Bag Reference Vector

In this section we will illustrate our bag-reference-based method applied to MIL problem. The miBRV aims to construct a vector representation for each bag by computing the similarity (distance in our method) with all other bags which are taken as the reference, transferring original features (with complex structure) into new bag-reference features containing rich information with simple linear structure. Our intuition is to use the distances with the reference bags to describe the bags’ intrinsic constitution and then train this affinity matrices to gain map function by a linear SVM.

3.1. Multi-instance Learning

Initially, we introduce the formal formulations of Multi-instance Learning. Given a data set of $\{(X_1, Y_1), \dots, (X_i, Y_i), \dots, (X_N, Y_N)\}$, where N is the number of bags, each bag X_i is consisted of grouped instances $(x_{i1}, \dots, x_{ij}, \dots, x_{in})$ and labeled with $Y_i \in \{1, -1\}$ while the instances’ labels are unknown. A positive bag contains at least one positive instance while there are only negative instances in negative bags. Thus, the task of Multi-instance Learning is to induce a classifier (or a mapping function) to predict the labels of input bags.

3.2. Bag Reference Vector

As what have been mentioned in section 1, intending to measure the relations between bags by means of set-to-set distances, we apply an operator to represent these distances. Hausdorff distance is a suitable technique to determine the extent to which one bag differs from another.

3.2.1 The Hausdorff Distance

Given two point sets $A = \{a_1, \dots, a_m\}$, $B = \{b_1, \dots, b_n\}$, the Hausdorff distance is defined as

$$H(A, B) = \max(h(A, B), h(B, A)),$$

where

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a_i - b_i\|.$$

Here the function $h(A, B)$, directed Hausdorff distance, is called forward Hausdorff distance from set A to

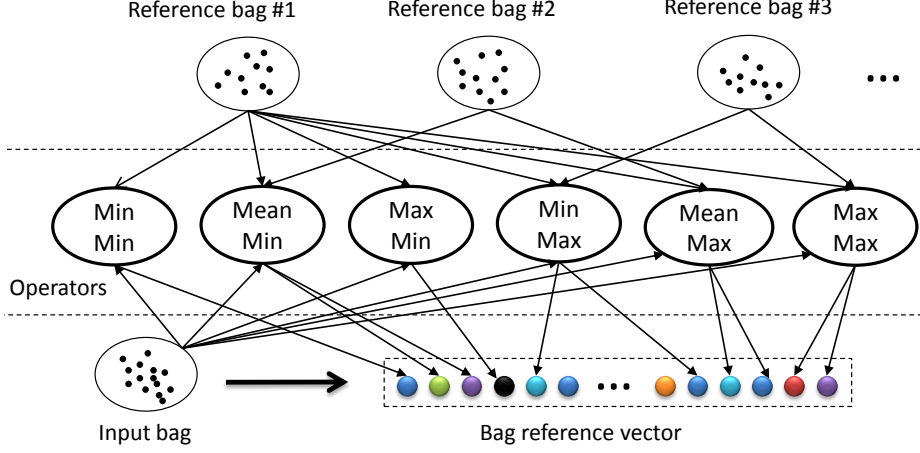


Figure 1. The illustration of how to build bag representation. The first row shows some reference bags; the second row shows some operators which take two bags and compute a distance; and the third row shows an input bag and its corresponding bag reference vector. Each dimension of the feature vector corresponds to a triplet {input bag, an operator, reference bag #n}.

set B as well. In addition, $\|a_i - b_i\|$ represents Euclidean distance between a_i and b_i , i.e., a point-to-point distance $d_{poi-poi}(a_i, b_i)$. For each $a_i \in A$, the algorithm will compute the point-point distance from b_1 to b_n and find the nearest point in set B to a_i with the least Euclidean distance, which is regarded as a point-set distance $d_{poi-set}(a_i, B)$. Hence, from set A to B , we are able to gain a distance vector $[d_{poi-set}(a_1, B), \dots, d_{poi-set}(a_m, B)]$ with m dimensions and vice versa as it is asymmetric distance. According to the definition for function h , the biggest one among these m shortest distances will be select as the value of $h(A, B)$, representing the distance from A to B . In this way we define a method to compute the set-to-set distance, in other words, this measures the similarity between set A and set B .

In terms of MIL, similar definition can be applied to it. Naturally, we treat each bag as a set like A or B and instances as the points. Thus, for bag X_i and X_j , we can apply forward Hausdorff distance to it as

$$h(X_i, X_j) = \max_{x_i \in X_i} \min_{x_j \in X_j} \|x_i - x_j\|$$

For bag X_i , we can obtain a bag-reference vector

$$b^{X_i} = [h(X_i, X_1), \dots, h(X_i, X_j), \dots, h(X_i, X_N)]$$

where $j = 1, 2, \dots, N$. In addition, b^{X_i} is l_2 normalized to reduce the influence of instance magnitude variation. With each bag's vector computed, an affinity matrices b^X can be extracted by means of Hausdorff distance. Then each bag b^X is delineated by being compared with the reference bags. The feature matrices are fed into a bag classifier along with bag labels for training and validation.

Pseudo code of miBRV is shown in Algorithm 1.

3.2.2 Extensions of Hausdorff Distance

The Hausdorff distance defines a point-to-set distance $d_{poi-set}$ by finding the nearest point in that set with the least Euclidean distance and then chooses the maximum among these point-to-set distances as final set-to-set distance. Thus, the operator is to obtain the maximal one among minimum values. This operation is suitable to most cases to gain correct descriptions for positive and negative bags.

Furthermore, considering the characteristic of MIL that some positive bags may well include negative instances, there will be some flaws in this algorithm for all multiple cases. For instance, if X_i is a positive bag with one negative instance x_{i1} while X_j is composed of positive instance only, after using Hausdorff distance, the bag-to-bag distance will be equal to the maximal instance-to-bag distance in $[d_{ins-bag}(x_{i1}, X_j), \dots, d_{ins-bag}(x_{im}, X_j)]$, which is $d_{ins-bag}(x_{i1}, X_j)$ as x_{i1} has the largest distance with all instances in X_j . This indicates using the Hausdorff distance between two positive bags results in choosing the distance from negative instance to positive bags to represent the similarity, which gives the misleading information. Consequently, modifications can be adopted in Hausdorff distance to gain several new affinity matrices as complements. Specifically, maximum, average and minimum operators have been added to enrich the distance definition to ameliorate incorrect representation. The following illustrates Hausdorff distance as well as other five distance measurement oper-

ators paralleled to it.

$$h_{minmin}^1(X_i, X_j) = \min_{x_{im} \in X_i} \min_{x_{jn} \in X_j} \|x_{im} - x_{jn}\|.$$

$$h_{meanmin}^2(X_i, X_j) = \text{mean}_{x_{im} \in X_i} \min_{x_{jn} \in X_j} \|x_{im} - x_{jn}\|.$$

$$h_{maxmin}^3(X_i, X_j) = h(X_i, X_j) = \max_{x_{im} \in X_i} \min_{x_{jn} \in X_j} \|x_{im} - x_{jn}\|.$$

$$h_{minmax}^4(X_i, X_j) = \min_{x_{im} \in X_i} \max_{x_{jn} \in X_j} \|x_{im} - x_{jn}\|.$$

$$h_{meanmax}^5(X_i, X_j) = \text{mean}_{x_{im} \in X_i} \max_{x_{jn} \in X_j} \|x_{im} - x_{jn}\|.$$

$$h_{maxmax}^6(X_i, X_j) = \max_{x_{im} \in X_i} \max_{x_{jn} \in X_j} \|x_{im} - x_{jn}\|.$$

Apart from extending the Hausdorff distance by adding more operators, many incorrect measured cases can be avoided by taking k nearest or farthest neighbors' average distance rather than just adopting one extreme case. To make it a practice, we firstly define two functions min_{-k} and max_{-k} which are computing the k_{th} largest and the k_{th} smallest distance respectively. To measure the distance between bag X_i and X_j , we can implement this addition to modify the function as follows.

$$\bar{h}_{minmin}^1(X_i, X_j) = \min_{x_{im} \in X_i} \frac{1}{k} \sum_{n=1}^k min_{-k} \|x_{im} - x_{jn}\|.$$

$$\bar{h}_{meanmin}^2(X_i, X_j) = \text{mean}_{x_{im} \in X_i} \frac{1}{k} \sum_{n=1}^k min_{-k} \|x_{im} - x_{jn}\|.$$

$$\bar{h}_{maxmin}^3(X_i, X_j) = \max_{x_{im} \in X_i} \frac{1}{k} \sum_{n=1}^k min_{-k} \|x_{im} - x_{jn}\|.$$

$$\bar{h}_{minmax}^4(X_i, X_j) = \min_{x_{im} \in X_i} \frac{1}{k} \sum_{n=1}^k max_{-k} \|x_{im} - x_{jn}\|.$$

$$\bar{h}_{meanmax}^5(X_i, X_j) = \text{mean}_{x_{im} \in X_i} \frac{1}{k} \sum_{n=1}^k max_{-k} \|x_{im} - x_{jn}\|.$$

$$\bar{h}_{maxmax}^6(X_i, X_j) = \max_{x_{im} \in X_i} \frac{1}{k} \sum_{n=1}^k max_{-k} \|x_{im} - x_{jn}\|.$$

The final representation of bag X_i , bag reference vector, is computed by a combination of these six distance operators, and denoted as

$$b^{X_i} = [\bar{h}^t(X_i, X_j)], \forall j \in [1, N], t \in [1, 6].$$

Combining all or some of these six distance operators will extract more distinctive features so that we can gain more comprehensive information for each bag with less training errors and improve the accuracy of the classifier.

3.3. Bag Classification using Linear SVM

As we can get a vector representation for a bag, we can use many existing classifiers for bag classification, such as SVM, Boosting, Random Forest. For efficiency, we use SVM with linear kernel for bag training and bag label prediction. The whole pipeline is illustrated in Algorithm 1. It consists with two steps, training and testing. For both training and testing, we use the LibLinear [8] toolbox.

Algorithm 1 miBRV for bag training and classification.

Input: Data set $\{(X_1, Y_1), \dots, (X_N, Y_N)\}$

TRAIN:

- 1: **for** $i = 1 \rightarrow N_{train}$ **do**
- 2: Map the original feature to Bag Reference Vector b^{X_i}
- 3: $b^{X_i} \leftarrow b^{X_i} / \|b^{X_i}\|_2$
- 4: **end for**
- 5: Use a linear SVM to train the transformed feature vectors $\{(b^{X_1}, Y_1), \dots, (b^{X_i}, Y_i), \dots, (b^{X_N}, Y_N)\}$ to learn a bag classifier B.

TEST:

- 6: **for** $j = 1 \rightarrow N_{test}$ **do**
- 7: Map the original feature to Bag-Reference Vector b^{X_j}
- 8: $b^{X_j} \leftarrow b^{X_j} / \|b^{X_j}\|_2$
- 9: **end for**

Output: The prediction of bag-level label $B(b^{X_j})$.

4. Experiments

4.1. Benchmark Data sets

In order to evaluate our method, we perform experiments on five benchmark data sets universally designed for MIL, including two Musk data sets [6] about molecule activity and three categories (elephant, fox, tiger) image data sets [1]. In details, there are 47 positive and 45 negative bags in Musk1 while Musk2 are composed of 39 positive and 63 negative bags which are described by conformations with 166-dimensional feature vector. On other three benchmark image data sets, each one is composed of 100 positive bags and 100 negative bags. We perform training and testing for ten times by 10-fold cross-validation, and average classification accuracy and standard deviation of each class are reported.

Several popular MIL algorithms including the state-of-the-arts: miFV [19], miGraph [24], MIBoosting, miSVM [1], EM-DD [23], and MIWrapper [7], are referred for comparison to evaluate our results. As shown in Table 1, it indicates that miBRV are so competitive

Table 1. The results on five benchmark data sets. The highest accuracies are highlighted in bold.

Algorithm	Musk1	Musk2	Elephant	Fox	Tiger	Average
miBRV	0.895 \pm 0.078	0.930 \pm 0.088	0.877 \pm 0.102	0.670 \pm 0.075	0.877 \pm 0.102	0.851
miFV[19]	0.909 \pm 0.089	0.884 \pm 0.094	0.852 \pm 0.081	0.621 \pm 0.109	0.813 \pm 0.083	0.816
miGraph[24]	0.889 \pm 0.073	0.903 \pm 0.086	0.869 \pm 0.078	0.616 \pm 0.079	0.801 \pm 0.083	0.816
MIBoosting [20]	0.837 \pm 0.120	0.790 \pm 0.088	0.827 \pm 0.073	0.638 \pm 0.102	0.784 \pm 0.089	0.775
miSVM [1]	0.874 \pm 0.120	0.836 \pm 0.088	0.822 \pm 0.073	0.582 \pm 0.102	0.789 \pm 0.089	0.781
EE-DD [23]	0.849 \pm 0.098	0.869 \pm 0.108	0.771 \pm 0.098	0.609 \pm 0.101	0.730 \pm 0.096	0.766
MIWrapper [7]	0.849 \pm 0.106	0.796 \pm 0.106	0.827 \pm 0.088	0.582 \pm 0.102	0.770 \pm 0.092	0.765

that it achieves the highest performance except on the MUSK1 data set. The average accuracy of miBRV over the five data sets has been improved by 3.5% to a large margin when comparing to the latest miFV method. The excellent results clearly demonstrate that miBRV is robust and can extract the most effective representation for a bag in MIL problems.

4.2. Text Categorization

Besides the benchmark tasks, the text categorization is another common application of MIL. For better comparison, we take the same twenty data sets derived from the 20 Newsgroups corpus as in [24]. In each category, there are 100 bags among which half bags are positive and others are negative. In addition, each instance is a post represented by the top 200 TF-IDF features.

In the same way, we carry out experiments on this data set using 10-fold cross-validation and report the average accuracy in Table 2. On this occasion, comparisons have been made between our miBRV, MI-Kernel and miGraph on these text categorization tasks. On 13 data sets out of 20, miBRV achieves the superior performance. The best average accuracy over all data sets indicates that the miBRV outperforms other two competing algorithms, miGraph and MI-Kernel [10].

4.3. Parameters Discussion

To deeply investigate miBRV, we discuss two main parameters in miBRV in this subsection. As illustrated in Section 3, we generate the final vector by combining different affinity matrices which are mapped by different distance operators together such as $[b_{h1}^X, \dots, b_{h6}^X]$ or just selecting some of them. In addition, the value of k , the number of averaging neighbors to be adopted, is a significant parameter for our experiment as well.

At first, we keep one factor, distance operator, unchanged with k changing from 1 to 4. The part of results for $[b_{h1}^X, b_{h2}^X, b_{h3}^X, b_{h4}^X, b_{h5}^X, b_{h6}^X]$ are shown in Table 3. These results illustrate that increasing k ameliorates the accuracy on Elephant and Fox but experiences a de-

Table 2. The results of twenty data sets of text categorization.

Data set	MIkernel	miGraph	miBRV
alt.atheism	60.2	65.5	77.0
comp.graphics	47.0	77.8	72.1
comp.os.ms-windows.misc	51.0	63.1	64.1
comp.sys.ibm.pc.hardware	46.9	59.5	69.0
comp.sys.mac.hardware	44.5	61.7	70.7
comp.window.x	50.8	69.8	80.7
misc.forsale	51.8	55.2	61.2
rec.autos	52.9	72.0	64.1
rec.motorcycles	50.6	64.0	54.4
rec.sport.baseball	51.7	64.7	77.8
rec.sport.hockey	51.3	85.0	85.0
sci.crypt	56.3	69.6	70.3
sci.electronics	50.6	87.1	90.7
sci.med	50.6	62.1	74.8
sci.space	54.7	75.7	67.8
sci.religion.christian	49.2	59.0	68.6
talk.politics.guns	47.7	58.5	66.2
talk.politics.mideast	55.9	73.6	65.1
talk.politics.misc	51.5	70.4	63.8
talk.religion.misc	55.4	63.3	60.8
Average	51.5	67.8	70.1

cline on Musk data sets at the same time. As a whole, it reaches the acme of average accuracy at $k=2$.

Then we fix k to 2 and test some different combinations of distance functions to extract a more informative feature vectors for diverse cases. Generally, higher dimensions feature vector improves the performance of classifier. And the more distance operators we used, the more robust miBRV is as a feature vector. Table 4 contains some details of the results with different parameters, from which we can find that, although results changes a lot from different parameters, most averaging accuracy are competitive with the state-of-the-art algorithms. The best performance of each column is bolded.

Table 3. The results on five benchmark data sets of parameter analysis. The distance operator is a combination of six operators.

Parameters	Musk1	Musk2	Elephant	Fox	Tiger
$k=1$	0.882 \pm 0.088	0.907 \pm 0.095	0.849 \pm 0.076	0.623 \pm 0.098	0.829 \pm 0.076
$k=2$	0.870 \pm 0.100	0.901 \pm 0.098	0.850 \pm 0.067	0.669 \pm 0.101	0.841 \pm 0.080
$k=3$	0.862 \pm 0.075	0.893 \pm 0.102	0.838 \pm 0.075	0.670 \pm 0.098	0.815 \pm 0.085
$k=4$	0.860 \pm 0.111	0.895 \pm 0.093	0.831 \pm 0.070	0.662 \pm 0.091	0.816 \pm 0.083

Table 4. The results on five benchmark data sets of parameter analysis. The value of k is 2.

Parameters	Musk1	Musk2	Elephant	Fox	Tiger
$[b_{h^1}^X, b_{h^3}^X]$	0.872 \pm 0.094	0.909 \pm 0.083	0.797 \pm 0.075	0.670 \pm 0.111	0.803 \pm 0.085
$[b_{h^2}^X, b_{h^5}^X]$	0.886 \pm 0.092	0.930 \pm 0.088	0.843 \pm 0.076	0.611 \pm 0.113	0.847 \pm 0.085
$[b_{h^2}^X, b_{h^4}^X, b_{h^5}^X]$	0.880 \pm 0.102	0.903 \pm 0.100	0.877 \pm 0.071	0.640 \pm 0.102	0.877 \pm 0.067
$[b_{h^1}^X, \dots, b_{h^6}^X]$	0.870 \pm 0.100	0.901 \pm 0.098	0.850 \pm 0.067	0.669 \pm 0.101	0.841 \pm 0.080

5. Conclusions

In this paper, we propose a novel technique for Multi-instance Learning. We focus on the inherent information on each bag, trying to delineate it by computing the similarity to other bags. In addition, our diverse distance definition fits it well, considering the crux of MIL that the proportion of positive instances in positive bags is ambiguous. No previous works adopt this straightforward but efficacious feature representation method. And the performances of our algorithm on these data sets popularly used for emulating MIL algorithms are superior to the state-of-the-art algorithms. What’s more, the proposed method produces a very simple vector representation for a bag, which works well with a linear SVM. Both the methodology and experimental results of the proposed approach show that it is very robust and effective.

In the future, on one hand, we may extend our method by changing the choice of reference bags. For instance, we can generate a great deal of reference bags in which the instances are randomly selected from the original bags. By this way we may describe our bags more accurately with more references if we can solve its possible computational expense. On the other hand, hewing to the intrinsic characteristic of MIL, we can extract features which describe the relationship of the instances in each bag by means of, for instance, adding some mathematical statistics such as standard deviation of instances in a bag, allowing us to distinguish different bags more clearly to solve the core problem of MIL tasks.

Acknowledgments

This work was primarily supported by National Natural Science Foundation of China (NSFC) (No. 61503145).

References

- [1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, 2002.
- [2] B. Babenko, N. Verma, P. Dollar, and S. Belongie. Multiple instance learning with manifold bag. In *ICML*, 2011.
- [3] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1619–1632, 2011.
- [4] R. G. Cinbis, J. Verbeek, and C. Schmid. Multi-fold mil training for weakly supervised object localization. In *CVPR*, 2014.
- [5] T. Deselaers and V. Ferrari. A conditional random field for multiple-instance learning. In *ICML*, 2010.
- [6] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1):31–71, 1997.
- [7] E.T.Frank and X.Xu. Applying propositional learning algorithms to multi-instance data. *University of Waikato, Department of Computer Science, University of Waikato, Hamilton, NZ, Tech. Rep.*, 2003.
- [8] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.
- [9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively

- trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.
- [10] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola. Multi-instance kernels. In *ICML*, 2002.
 - [11] D. P. Huttenlocher, G. Klanderman, W. J. Rucklidge, et al. Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, 1993.
 - [12] G. Liu, J. Wu, and Z.-H. Zhou. Key instance detection in multi-instance learning. 2012.
 - [13] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *ICML*, 1998.
 - [14] D. Parikh and K. Grauman. Relative attributes. In *ICCV*, 2011.
 - [15] W. Shen, B. Wang, Y. Wang, X. Bai, and L. J. Latecki. Face identification using reference-based features with message passing model. *Neurocomputing*, 99:339–346, 2013.
 - [16] P. Viola, J. C. Platt, and C. Zhang. Multiple instance boosting for object detection. In *NIPS*, 2006.
 - [17] X. Wang, Z. Zhang, Y. Ma, X. Bai, W. Liu, and Z. Tu. One-class multiple instance learning via robust pca for common object discovery. In *ACCV*. 2013.
 - [18] X. Wang, Z. Zhang, Y. Ma, X. Bai, W. Liu, and Z. Tu. Robust subspace discovery via relaxed rank minimization. *Neural computation*, 26(3):611–635, 2014.
 - [19] X.-S. Wei, J. Wu, and Z.-H. Zhou. Scalable multi-instance learning. In *ICDM*, 2014.
 - [20] X. Xu and E. Frank. Logistic regression and boosting for labeled bags of instances. In *Knowledge Discovery and Data Mining*, pages 272C–281. Proc. 8th Pacific-Asia Conf, 2004.
 - [21] Z.-J. Zha, X.-S. Hua, T. Mei, J. Wang, G.-J. Qi, and Z. Wang. Joint multi-label multi-instance learning for image classification. In *CVPR*, 2008.
 - [22] K. Zhang and H. Song. Real-time visual tracking via online weighted multiple instance learning. *Pattern Recognition*, 46(1):397–411, 2013.
 - [23] Q. Zhang and S. A. Goldman. Em-dd: An improved multiple-instance learning technique. In *NIPS*, 2001.
 - [24] Z.-H. Zhou, Y.-Y. Sun, and Y.-F. Li. Multi-instance learning by treating instances as non-iid samples. In *ICML*, 2009.